

# A hierarchical, automated target recognition algorithm for a parallel analog processor

Curtis Padgett

Gail Woodward

Jet Propulsion Laboratory  
California Institute of Technology

A hierarchical approach is described for an automated target recognition (ATR) system, VIGILANTE, that uses a massively parallel, analog processor (3D ANN). The 3D ANN processor is capable of performing 64 concurrent inner products of size  $1 \times 4096$  every 250 nanoseconds. A complete  $64 \times 64$  raster scan of a  $256 \times 256$  image can be evaluated by the 3D ANN with its 64 modifiable templates in about 16 milliseconds. To fully utilize the analog processor and accommodate its high bandwidth requirements, the vectors (templates) loaded on to the 3D ANN provide dimensionality reduction for a back-end set of classifiers. The templates used in the reported algorithm are hierarchically generated sets of eigenvectors taken from a partitioned set of library object images. As information is accumulated about the target (e.g. object class), a more refined set of eigenvectors reflecting this knowledge can be loaded and more specialized classifiers utilized. The classifiers provide information related to the ATR task: location, class, sub-class, and orientation of target(s). We report some preliminary results that examine the performance of orientation classifiers. With no knowledge about object class or orientation, a neural network achieves 94.2% in determining to which one of three classes from vertical ( $30^\circ$ ,  $45^\circ$ , or  $60^\circ$ ) an object image is oriented ( $\pm 30^\circ$ ). Using an eigenvector template set generated from a distribution where both object class and orientation are known, a neural network classifier achieves 96% in orienting the object to within  $\pm 22.5^\circ$ . This information can be used to load even more specific eigenvector sets which should lead to more accurate object location during tracking and an enhancement in object recognition tasks.

## 1 Introduction

Many of the problems associated with automated target recognition (ATR) have been studied independently but no fully automated, general imaging system exists that is able to detect an aerial target, distinguish it as hostile (or interesting), and then perform real time tracking. The reasons for this are quite well known in the machine vision community. For most non-trivial targets, changes in appearance due to lighting variations, scale, orientation, clutter, etc. pose significant difficulties in determining where an object is, its class, sub-class, or its orientation. These difficulties often require high resolution imagery and substantial amounts of computation to successfully evaluate a scene. Previously, only large supercomputers were able to provide the processing power to run most ATR algorithms at anything close to the frame rates sufficient for tracking a fast airborne target.

The VIGILANTE project is currently developing a gimbaled, optical bench coupled to a massively, parallel processing engine designed to perform as a self-contained ATR system on an airborne platform. The gimbaled optical bench (VIGIL) provides a multiple sensor imaging system that delivers a single  $256 \times 256$  image from one of the sensors to a high speed data processing path (ANTE) at 30 frames per second. The set of sensors to be integrated into the optical bench include four sensors: a 1.5 degree field of view (FOV) CCD; a CCD camera with a controllable zoom; an infra-red QIP camera; and an ultra-violet camera. The optical bench is controlled by

a host computer (a P6 running at 200 MHz) that selects the active sensor, directs the gimbal, adjusts the zoom, and sets other sensor specific parameters (e.g. exposure time).

The data processing path of VIGILANTE, provides a dedicated set of digital and analog parallel processors to implement the high bandwidth (30 frames per second) imaging operations required for ATR applications. The heart of the ANTE system is the 3D ANN processor. It is an analog processor, capable of performing 64 concurrent vector dot product operations of 4096 dimensions each, every 250 nanoseconds. A back end digital parallel machine consisting of 512 processors receives the output from the 3DANN, performs simple parallel operations (e.g. a neural network classifier) and sends its results to the host processor for final evaluation.

The 3DANN effectively performs 64 convolutions on the original image with 64x64 templates in approximately 16 milliseconds. The job of the digital post-processor and the host is to distill the information extracted by the 3D ANN and put it in a form suitable for use by an ATR application: target classification, type, identification, direction, etc. The next section describes the 3DANN architecture more fully. We then describe our proposal for controlling the VIGILANTE machine during an ATR task. Finally, we show preliminary results using neural network classifiers to provide orientation information within our control framework.

## 2 Data Processing Path

The data processing path of VIGILANTE consists of a frame grabber, a digital loading device and digital to analog converter (CLIC), the 3DANN processor and a bank of analog to digital converters, the SIMD, 512 processor CNAPS boards, and the host P6 with its secondary storage. The key components of the system are shown in Figure 1. The frame grabber stores the image from the active sensor on the optical bench and moves it in 1x64 pixel rows or columns to the CLIC. The CLIC takes in a row or column and shifts it on to its 64x64 array of digital to analog converters each clock cycle (250 nanoseconds). The 64x64 digital elements are converted to analog and placed on the 3D ANN's internal bus for inner product calculation.

The central component of VIGILANTE is the 3D ANN module. It has 64 64x64 digitally specified weight templates. These templates and the analog input signal from the CLIC are evaluated every 250 nanoseconds resulting in a 64 dimensional output vector,  $\mathbf{v}$ -

$$v_i = \sum_{j=1}^{64 \times 64} c_j * T_{i,j}$$

where  $\mathbf{c}$  is a 64x64 input image and  $\mathbf{T}$  is the matrix defined by the templates. The templates in the 3D ANN module are specified with 8 bit precision. It takes approximately 1 millisecond to load a new set of templates provided the set was resident in the loader's buffer (the loader provides space for up to 5 sets of templates). If the templates need to be retrieved from the host, the time it takes to load the templates is between 25-50 milliseconds (25 if the templates are resident in main memory). A complete convolution of the 256x256 image with a 64x64 template requires approximately 16 milliseconds to complete.

The 64 analog values generated by the 3D ANN and are converted to 8 bit digital values and are passed along to a memory buffer where they can be accessed by the CNAPS processors (Adaptive Solutions, Beaverton, Oregon). The boards process the output of the 3D ANN and reduce the amount of information (64- 256x256 images) so that it can be efficiently evaluated by the host. The host processor selects the sensor, moves templates on and off the 3DANN, selects which algorithms run on the CNAPS boards, and evaluates the results of the data processing. It also provides "context," for carrying out the ATR tasks as it remembers which target is being tracked and updates VIGILANTE's state appropriately to reflect this.

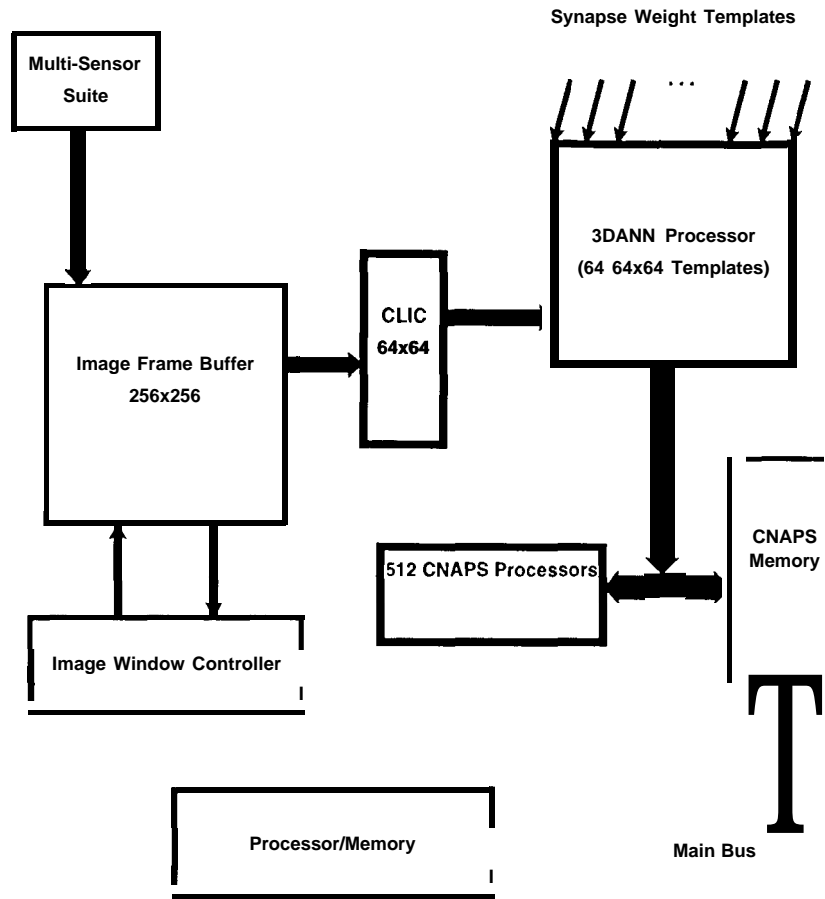


Figure 1: VIGILANTE architecture.

### 3 Algorithm Description

Our goal for VIGILANTE is to demonstrate that the system is able to acquire an airborne object, and subsequently recognize and track the target over multiple frames. The variations in a target's appearance due to class (helicopter, missile, plane, etc.), identity, orientation, lighting, and scale make this task extremely difficult. Simple correlation schemes (each template is an object image) using mismatch energy are bound to fail due to the extremely large number of templates required to accurately depict the appearance of each target for classification. For instance with 3 classes each with a single object, 4 scale sizes, 4 illumination schemes, and 200 object orientations, over 9000 templates would have to be processed and the results evaluated prior to determining if a particular object is in the frame.

In a ATR scenario, even an extremely fast processor like the 3DANN would be unable to search the 9000 templates quick enough to make real time decisions. Obviously, a different type of search strategy is required for ATR type applications. Modification of correlation strategies to allow for composite templates, reducing the dimensionality of the image using either wavelets or eigenvectors, and the use of sensor fusion are all techniques that have been applied to the ATR problem.<sup>9,1,7,10,6</sup>

In VIGILANTE, we approach the ATR problem using a hierarchical methodology. Our knowledge base consists

## Hierarchical Object Recognition

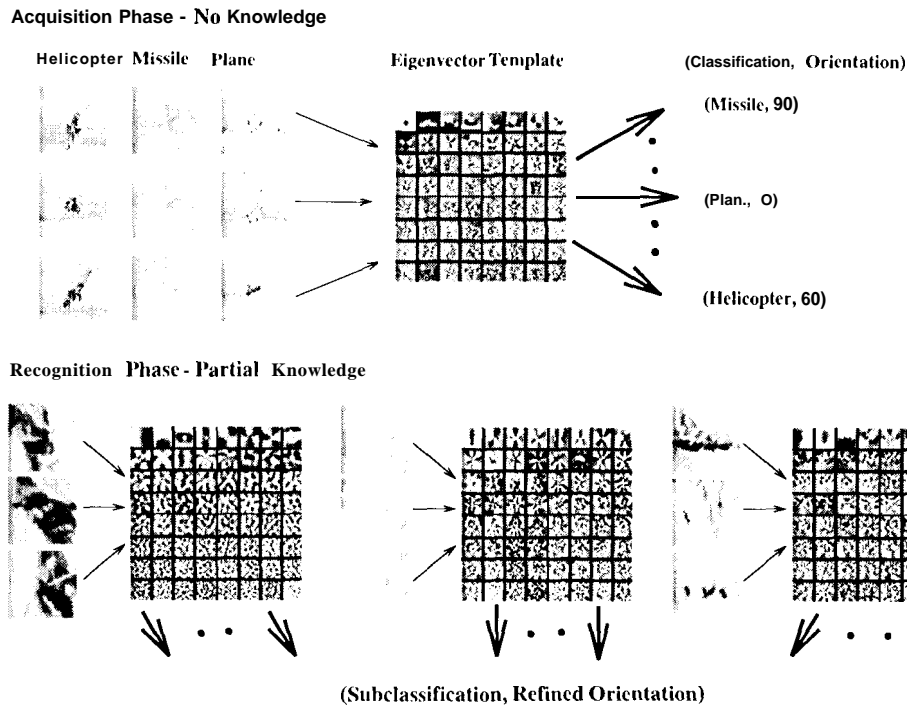


Figure 2: Example imagery from the object libraries and corresponding eigenvector sets for the template hierarchy. The template set that corresponds to VIGILANTE's current knowledge about the target is loaded on the 3DANN and associated classifiers are run concurrently on the CNAP's processors during frame evaluation,

of a set of object libraries that give the appearance of an object for various orientations, lighting conditions, and scale. Sets of templates are generated with principal components analysis (PCA) by sampling the library images at different levels of specificity with respect to object class, sub-class, and orientation. PCA provides an ordered set of eigenvectors that have been shown to be useful for face and object recognition and identification tasks. § 3.4 The most general templates are generated using PCA from a random sample drawn over the entire database. More specific template sets are generated and reflect system knowledge of the object class, sub-class, and orientation. These variables are explicitly evaluated for during image processing over the course of an ATR scenario.

The motivation for using a hierarchy is straight forward. Some information about a target is difficult to ascertain and may take many frames or an appropriate view in order to obtain it. Problems such as identification may require fine distinctions between closely related objects. Using a hierarchical set of eigenvectors allows the system to perform *easier* tasks early with projections on less specific eigenvector sets. The answers to these questions (class, orientation within 90°) can be used to narrow the distribution of expected object images to a single class or subclass at a more specific orientation. The eigenvectors of such a distribution will provide finer discrimination between object appearances thus allowing even more subtle characterization (e.g. identity) to be made. Classifiers at each stage of the processing are working with an eigenvector set where the distribution is maximal for the object of interest. Figure 2 shows the template hierarchy and the decision variables used to update the resident template set.

The algorithm we have developed for VIGILANTE is organized in three phases: Acquisition, Recognition, and Tracking. During each phase, neural networks (or possibly other classifiers) trained on the projected values

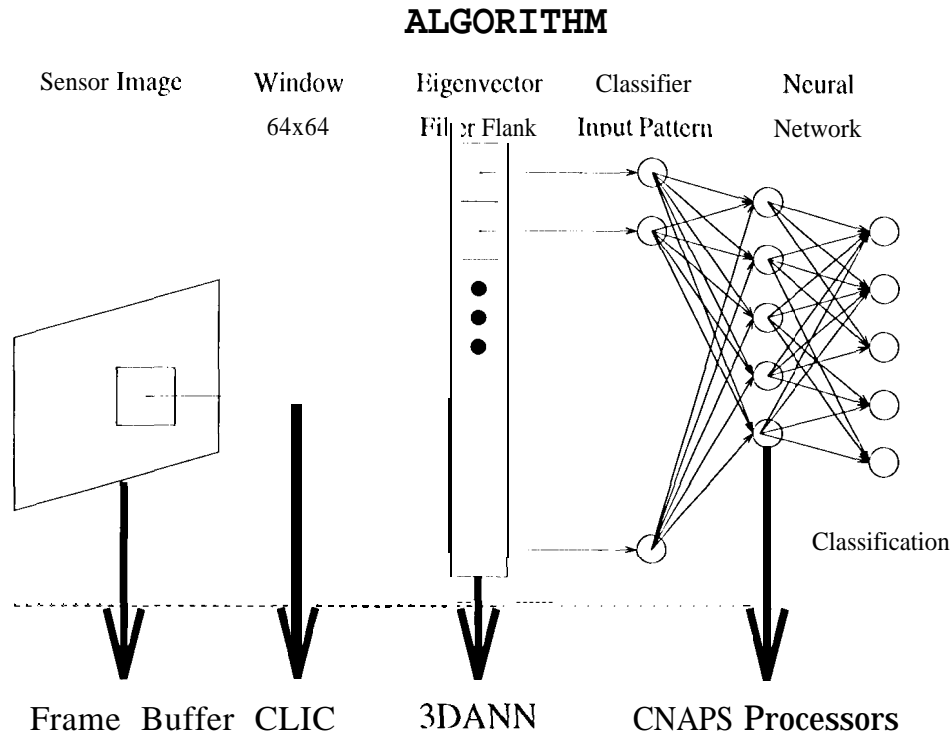


Figure 3: Algorithm to VIGILANTE architecture.

from the template set, locate the object (or its features) in the frame and provide more specific orientation and recognition information. In the Acquisition Phase (AP) it is assumed that a airborne target's image is of the appropriate size (greater than 20 pixels if seen at profile for the given set of optics) if it is in the current frame. No other knowledge is assumed about the target. The work required in this phase is to locate the object within the frame, determine the class of the object (helicopter, plane, or missile), and establish a rough orientation for the principal axis of the object (within  $\pm 45^\circ$ ). In the Recognition Phase (RP), the eigenvectors specific to object class and orientation (as determined in AP or prior evaluation in RP) are loaded onto the 3DANN. As information about subclass, identity, and orientation becomes known about the target, the appropriate template set is loaded on the 3DANN. This phase is continued until the object is completely identified and a decision has been made to track it. The Tracking Phase ('T') is designed to continuously maintain object tracking capability when the object is too large for centroid tracking. In this event, a feature tracking system is implemented to provide additional orientation and targeting information.

Figure 3 shows the mapping of the image evaluation algorithm on the VIGILANTE architecture during AP and RP. The 3DANN projects a sub-window in the current frame on each of its 64 templates every 250 nanoseconds. The CLIC is used to select the window to be evaluated and performs the necessary digital to analog conversion. The CNAPS boards implement neural network classifiers which take the 3DANN output (the 64 projected values) and implement one of three specific tasks in parallel: location) classification, and orientation.

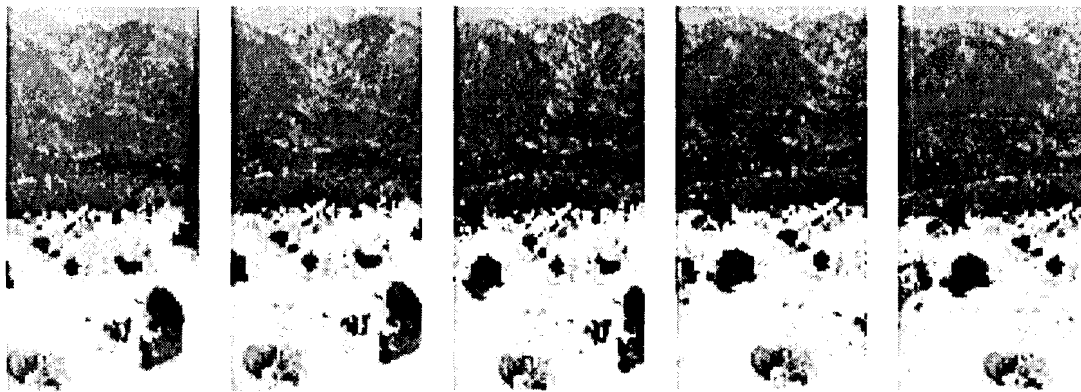


Figure 4: Example imagery used for testing orientation. The object is a model L-raise missile.

## 4 Results

This section describes simulation results for orienting the principal axis of the object in the image plane using the hierarchical methodology. To reduce the complexity of the problem a number of simplifying assumptions are made. Only 3 classes of objects, each containing a single object sub-class are considered helicopters, cruise missiles, and planes. The object is centered in the frame (all evaluated images have objects) using the output of an object detector. The object measures at least 30 pixels in profile at the distance imaged along its principal axis (independent of the object). Two scales are incorporated in the database.

An object library was constructed with three models (one from each class). The objects were oriented against a plain background and imaged with diffuse lighting. Image operations were performed on the data to account for scale and some orientations. For testing purposes, the models were imaged in a heavily cluttered scene. Figure 4 provides typical examples of test imagery.

Two experiments on orientation were performed using the test imagery, simulating the hierarchical approach described in the previous section. In the first experiment, no knowledge was assumed about the object. The task was to determine which orientation as measured from perpendicular-  $30^\circ$ ,  $45^\circ$ , or  $60^\circ$  best describe the given object. The classifier was considered successful if the true angle of the principal axis of the object was within  $\pm 30^\circ$  of the labeled class. The eigenvector set used to generate the 64 values associated with the image were generated from a sample of 1000 images drawn randomly from the entire object library.

In the second experiment, the class of the object and its orientation to within  $\pm 45^\circ$  is given. In this case, the classifier was to refine the orientation estimate. To be successful, it needed to be within  $\pm 22.5^\circ$  of the true orientation. As object class was known, three eigenvector sets were generated from a random sample of 1000 images drawn using the identified object over a uniform range of allowed orientation. Figure 2 provides images of the actual eigenvector sets used in the study.

The orientation classifier is a simple feed forward neural network. It employs a single hidden layer with 20 nodes which feeds a single output variable taking on values between  $\pm 1.0$ . Each sub-image block is evaluated independently by first projecting it on each of the 64 eigenvectors and then providing these outputs to the neural network for classification. The output is then linearly mapped back to an angle. The network is trained with back propagation<sup>2</sup> on images in the object libraries incorporating scale, object, and orientation variations. A portion of the training set (a hold out set) is reserved to stop network learning in order to enhance generalization. Also included in the training and hold out set are some examples of the object in a cluttered environment. This was

## Image Frame Distribution Set

Output Response: 1

0

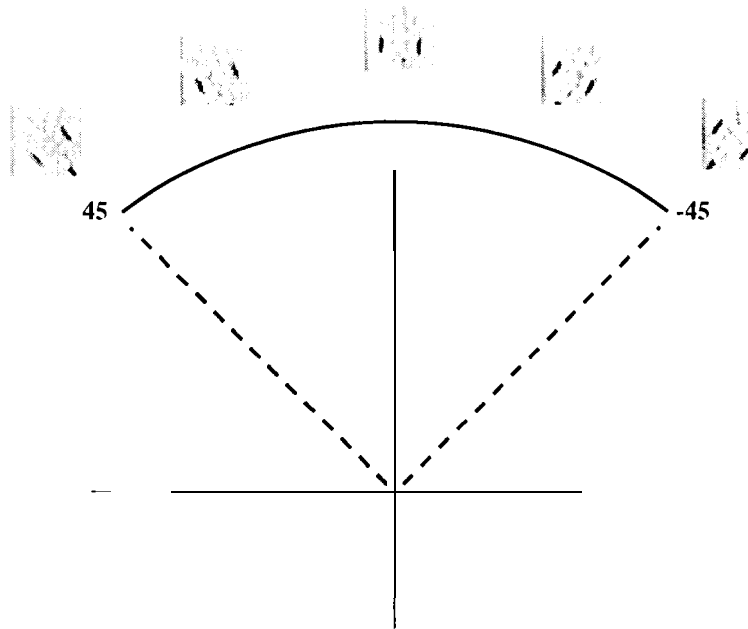


Figure 5: Neural network output mapping for image plane orientation.

done to enhance generalization for the test imagery.

Figure 6 presents the results of the classifiers on the respective objects. The neural networks had little difference in their identification rates with respect to object class. Only the missile orientation for the missile was significantly higher in the Acquisition Phase. This is most likely due to its relatively simple shape (as compared to the helicopter or plane) and its brightness when compared with background. Most likely higher results can be obtained with more precise data. The object libraries were generated using manual alignment. We are currently installing a totally automated system for generating the object libraries from scale models.

## 5 Conclusion

We have described an algorithm that is easily implemented in the VIGILANTE architecture that provides good orientation rates on three airborne model objects in a cluttered scene. The algorithm provides a straight forward hierarchical decomposition of an ATR scenario general template sets for easy tasks when little is known about the object, more specific template sets for the finer discrimination needed for recognition or identification.

<i>Object</i>	<i>Phase</i>	
	Acquisition	Recognition
missile	96.3	95.4
plane	93.3	96.0
helicopter	93.1	95.6

Figure 6: Evaluation success rate for determining the image plane orientation of an object. The rate is the percentage of novel images whose neural network output was linearly mapped to within 45.0 or 22.5 degrees of the actual orientation of the principal axis in the image plane depending up on which phase (Acquisition or Recognition) was being evaluated. For each object, approximately 6000 images were used to estimate the rate.

## 6 REFERENCES

- [1] M.C. Burl, U. Fayyad, P. Perona, P. Smyth, and M.J. Burl. Automating the hunt for volcanoes on venus. In *Proceedings of the 1994 IEEE Computer Vision and Pattern Recognition Conference*, Seattle, WA, 1994.
- [2] James J. McClelland, David E. Rumelhart, and the PDP Research Group. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 2. The MIT Press, Cambridge, 1986.
- [3] H. Murase and S. Nayar. Learning and recognition of 3d objects from appearance. In *IEEE 2nd Qualitative Vision Workshop*, New York, June 1993. IEEE Press.
- [4] C. Padgett and G. Cottrell. Representing face images for classifying emotions. In *Advances in Neural Information Processing Systems 9*, Cambridge, MA, 1997. MIT Press.
- [5] Curtis Padgett, M. Zhu, and S. Suddarth. Detection and orientation classifier for the vigilante image processing system. in *SPIE International Symposium on Aerospace/Defense Sensing and Dual-Use Photonics*, 1997.
- [6] L. Perlovsky, J. Chernick, and W. Schoendorf. Multi-sensor atr and identification of friend or foe using mlans. *Neural Networks*, 8(7):1185-1200, 1995.
- [7] S. Rogers, J. Colombi, C. Martin, J. Gainey, K. Fielding, P. Burns, D. Ruck, hf. Kabrisky, and M. Oxley. Neural networks for automatic target recognition. *Neural Networks*, 8(7):1153-1184, 1995.
- [8] Matthew Turk and Alexander Pentland. Eigenfaces for recognition. *The Journal of Cognitive Neuroscience*, 3:71-86, 1991.
- [9] B.V. K. Vijaya Kumar. Tutorial survey of composite filter designs for optical correlators. *Applied Optics*, 31(23):4773-4801, Aug 1992.
- [10] A. Waxman, M. Seibert, A. Gove, D. Fay, A. Bernardon, C. Lazott, W. Steele, and Cunningham R. Neural processing of targets in visible, multispectral ir and sar imagery. *Neural Networks*, 8(7):1029-1051, 1995.